

## EnergyPlus: creating a new-generation building energy simulation program

Drury B. Crawley<sup>a,\*</sup>, Linda K. Lawrie<sup>b</sup>, Frederick C. Winkelmann<sup>c</sup>, W.F. Buhl<sup>c</sup>,  
Y. Joe Huang<sup>c</sup>, Curtis O. Pedersen<sup>d</sup>, Richard K. Strand<sup>d</sup>, Richard J. Liesen<sup>d</sup>,  
Daniel E. Fisher<sup>e</sup>, Michael J. Witte<sup>f</sup>, Jason Glazer<sup>f</sup>

<sup>a</sup>US Department of Energy, EE-41, 1000 Independence Avenue, SW, Washington, DC 20585-0121, USA

<sup>b</sup>US Army Engineer Research and Development Center, Construction Engineering Research Laboratory, Champaign, IL 618, USA

<sup>c</sup>Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

<sup>d</sup>University of Illinois at Urbana-Champaign, Urbana, IL 61801-1867, USA

<sup>e</sup>Oklahoma State University, Oklahoma, OK 74078, USA

<sup>f</sup>GARD Analytics, Park Ridge, IL 60068, USA

---

### Abstract

Many of the popular building energy simulation programs around the world are reaching maturity — some use simulation methods (and even code) that originated in the 1960s. For more than two decades, the US government supported development of two hourly building energy simulation programs, BLAST and DOE-2. Designed in the days of mainframe computers, expanding their capabilities further has become difficult, time-consuming, and expensive. At the same time, the 30 years have seen significant advances in analysis and computational methods and power — providing an opportunity for significant improvement in these tools.

In 1996, a US federal agency began developing a new building energy simulation tool, EnergyPlus, building on development experience with two existing programs: DOE-2 and BLAST. EnergyPlus includes a number of innovative simulation features — such as variable time steps, user-configurable modular systems that are integrated with a heat and mass balance-based zone simulation — and input and output data structures tailored to facilitate third party module and interface development. Other planned simulation capabilities include multizone airflow, and electric power and solar thermal and photovoltaic simulation. Beta testing of EnergyPlus began in late 1999 and the first release is scheduled for early 2001. © 2001 Elsevier Science B.V. All rights reserved.

**Keywords:** Building performance; Simulation; Energy performance; Heat balance; Mass balance; Modular simulation

---

### 1. Introduction

For more than 20 years, the US government supported development of two building energy simulation programs, DOE-2 and BLAST. BLAST [1], sponsored by the US Department of Defense (DOD), has its origins in the NBSLD program developed at the US National Bureau of Standards (now NIST) in the early 1970s. DOE-2 [2], sponsored by the US Department of Energy (DOE), has its origins in the Post Office program written in the late 1960s for the US Post Office. The main difference between the programs is load calculation method — DOE-2 uses a room weighting factor approach while BLAST uses a heat balance approach. Both programs are widely used throughout the world.

Each program comprises hundreds of subroutines working together to simulate heat and mass energy flows throughout a building. In some cases, subroutines in DOE-2 were more accurate. In other cases, subroutines in BLAST were more accurate. In both cases, simulation methodologies are often difficult to trace due to decades of development (and multiple authors). Often, this results in ‘spaghetti code’ with data and subroutines for a particular simulation capability spread throughout the program. To modify either program, a developer must have many years experience working within the code, knowledge of code unrelated to their task (because of the spaghetti), and (often for sponsors) an extraordinary investment of time and money.

Many people questioned why the US government was supporting two separate (and comparable capability) programs. Discussions on merging the two programs began in earnest in 1994 with a DOD-sponsored workshop. Although nothing concrete resulted from that

---

\* Corresponding author. Tel.: +1-202-586-2344; Fax: +1-202-586-5557.  
E-mail address: drury.crawley@ee.doe.gov (D.B. Crawley).

workshop, DOE eventually took the initiative and began developing a new program named EnergyPlus in 1996. The EnergyPlus team includes US Army Construction Engineering Research Laboratories (CERL), University of Illinois (UI), Lawrence Berkeley National Laboratory (LBNL), Oklahoma State University (OSU), GARD Analytics, and DOE. In this paper, we present an overview of the organization and capabilities of EnergyPlus and explain the rationale and structure behind the overall program.

### 1.1. What is EnergyPlus?

As mentioned earlier, DOE-2 and BLAST have become expensive to maintain, modify and enhance — because of 20+ year old code and old Fortran structures (or general lack of structure). When DOD ended support for BLAST in 1995 due to budget constraints, we took the opportunity to combine the resources, teams, and best capabilities and

features of BLAST and DOE-2. The last version of BLAST was released in spring 1998 and the last version of DOE-2 with contributions under DOE-sponsorship was released in spring 1998. Initially, we thought that we could create a ‘best of’ program — combining modules from the two programs — without starting from scratch. After initial development work, we determined that EnergyPlus would cost less to develop, be released faster, and be easier to modify and extend if we wrote all new code.

Thus, EnergyPlus is an all-new program, albeit one based on the most popular features and capabilities of BLAST and DOE-2. EnergyPlus comprises completely new, modular, structured code written in Fortran 90. It is primarily a simulation engine — there is no interface. Fig. 1 shows screen images of EnergyPlus utilities and example results. Input and output are simple comma-separated, ASCII text files, much simpler input than either DOE-2 or BLAST. Both BLAST and DOE-2 have successfully attracted third-party

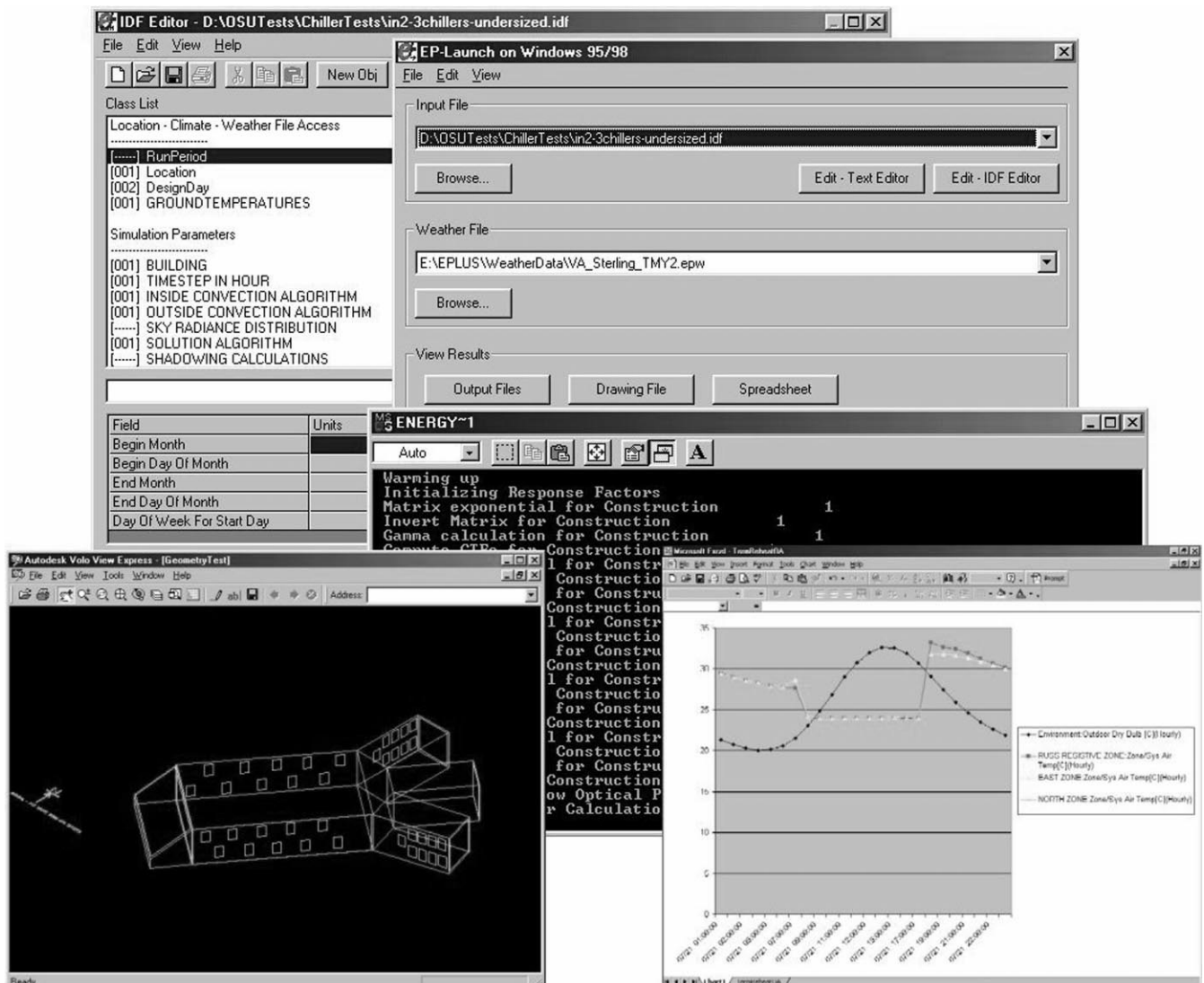


Fig. 1. Screen images of EnergyPlus utilities and results.

developers to create user interfaces and new modules. During EnergyPlus beta testing we invited these same developers to work with the development team on new simulation modules or develop their own user interfaces.

### 1.2. Modular code

One of the main goals we set for developing EnergyPlus was to create a well-organized, modular structure to facilitate adding features and links to other programs. In evaluating programming languages, we found we had two choices — move to C/C++ or stay with Fortran. Despite the advantages of the structure and object-orientation of C/C++, we decided to select Fortran 90 as the programming language for EnergyPlus because Fortran 90:

- is a modern, modular language with good compilers on many platforms;
- allows C-like data structures and mixed language modules;
- provides structure that begins to be object-based;
- allows long variable names (up to 32 characters);
- provides backward compatibility during the development process.

We began working on EnergyPlus by modularizing (restructuring) code from the heat balance engine in IBLAST, a research version of BLAST with integrated loads and HVAC calculation [3,4]. Normally such restructuring would result in major rewrites involving a long development period, and extensive testing to ensure new code performs as intended. However, because the EnergyPlus team selected Fortran 90 (and Fortran 77 is a subset of Fortran 90), development proceeded through a process which we call evolutionary reengineering (ER). This process incrementally moves the program from old unstructured legacy code to new modular code by incorporating new code with old. The existing code still works with user input data, and is extended to generate parameters needed by the new code modules. In this way, new modules were verified without having to completely replace the entire functional capability of the old program with new code before it was tested. As the process proceeds, the parameters supplied by old routines are replaced by new routines and data structures. This makes the transition evolutionary and permits a smooth transition with a greater capability for verification testing.

## 2. EnergyPlus structure

In two recent workshops on next generation energy tools sponsored by DOE and DOD [5], there was strong consensus that a more flexible and robust tool with additional capabilities is needed. Recurrent simulation needs throughout both workshops were design, environment, economics, and occupant comfort and safety. Designers need tools that provide answers to very specific questions during design.

They want tools that provide the highest level of simulation accuracy and detail reasonably possible, but that do not get in the user's way. One of the highest priorities was an integrated (simultaneous loads and systems) simulation for accurate temperature and comfort prediction.

In response to these findings, we decided that integrated simulation should be the underlying concept for EnergyPlus. Loads calculated (by a heat balance engine) at a user-specified time step (15-min default) are passed to the building systems simulation module at the same time step. The building systems simulation module, with a variable time step (down to seconds), calculates heating and cooling system and plant and electrical system response. Feedback from the building systems simulation module on loads not met is reflected in the next time step of the load calculations in adjusted space temperatures if necessary.

By using an integrated solution technique in EnergyPlus, the most serious deficiency of the BLAST and DOE-2 sequential simulations can be solved — inaccurate space temperature prediction due to no feedback from the HVAC module to the loads calculations. Accurate prediction of space temperatures is crucial to energy efficient system engineering — system size, plant size, occupant comfort and occupant health are dependent on space temperatures.

Integrated simulation also allows users to evaluate a number of processes that neither BLAST nor DOE-2 can simulate well. Some of the more important include:

- realistic system controls;
- moisture adsorption and desorption in building elements;
- radiant heating and cooling systems;
- interzone air flow.

Fig. 2 shows the overall program structure. EnergyPlus has three basic components — a simulation manager, a heat and mass balance simulation module, and a building systems simulation module. The simulation manager controls the entire simulation process. The heat balance calculations are based on IBLAST — a research version of BLAST with integrated HVAC systems and building loads simulation.

A new building systems simulation manager handles communication between the heat balance engine and various HVAC modules and loops, such as coils, boilers, chillers, pumps, fans, and other equipment/components. (In the first release, the building systems simulation manager only has HVAC systems and equipment/components. Future releases of EnergyPlus will include electrical systems simulation.) The building systems simulation manager also controls interaction and data exchange between EnergyPlus and SPARK [6] and TRNSYS [7] simulations. Gone are the hardwired 'template' systems (VAV, constant volume reheat, etc.) of DOE-2 and BLAST — replaced by user-configurable heating and cooling equipment components formerly within the template. This gives users much more flexibility in matching their simulation to the actual system configurations. The building systems simulation module also

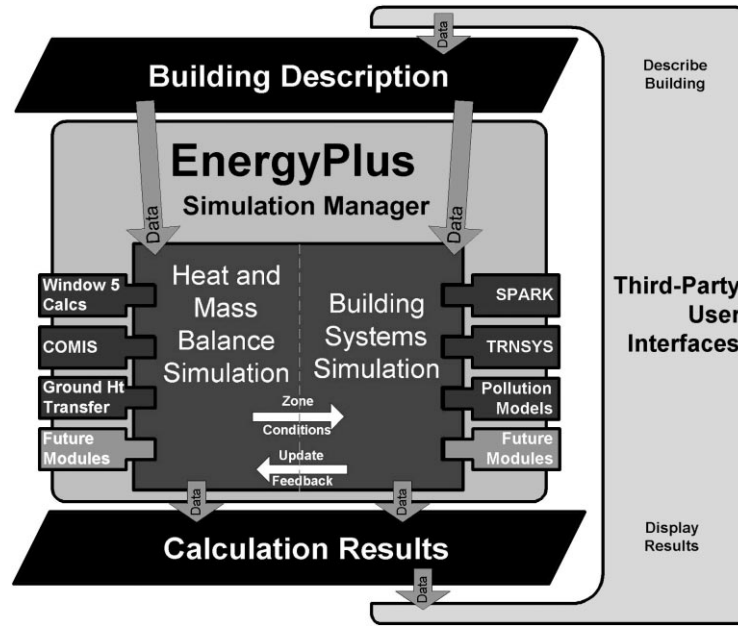


Fig. 2.

manages data communication between the HVAC modules, input data, and output data structures.

A comparison of major features and capabilities of EnergyPlus, BLAST, IBLAST, and DOE-2 is shown in Tables 1–3. Table 1 shows general features, Table 2 load calculation features, and Table 3 HVAC features. The simulation manager, heat balance simulation manager, and building systems simulation manager are described in more detail below.

### 2.1. Simulation management

At the outermost program level, the simulation manager controls the interactions between all simulation loops from a sub-hour level up through the user selected time step and

simulation period — whether day, month, season, year or several years. Actions of individual simulation modules are directed by the simulation manager, instructing simulation modules to take actions such as initialize, simulate, record keep, or report.

We created the simulation manager to specifically address the legacy issues of spaghetti code and lack of structure in DOE-2 and BLAST. The simulation manager provides several critical benefits:

- major simulation loops are contained in a single module;
- modules are self-contained and more object-based;
- data access is controlled;
- new modules are easily added.

Table 1  
Comparison of general features and capabilities

General feature	DOE-2	BLAST	IBLAST	EnergyPlus
Integrated, simultaneous solution	No	No	Yes	Yes
Integrated loads/systems/plant				
Iterative solution				
Tight coupling				
Multiple time step approach	No	No	Yes	Yes
User-defined time step for interaction between zones and environment (15-min default)				
Variable time-step for interactions between zone air mass and HVAC system ( $\geq 1$ min)				
Input functions	Yes	No	No	Yes
Users can modify code without recompiling				
Reporting mechanism				
Standard reports	Yes	Yes	Yes	Yes
User-definable reports	Yes	No	No	Yes
Visual surface output	No	No	No	Yes

Table 2  
Comparison of loads features and capabilities

Loads feature	DOE-2	BLAST	IBLAST	EnergyPlus
Heat balance calculation	No	Yes	Yes	Yes
Simultaneous calculation of radiation and convection processes each time step				
Interior surface convection				
Dependent on temperature and air flow	No	Yes	Yes	Yes
Internal thermal mass	Yes	Yes	Yes	Yes
Moisture absorption/desorption	No	No	Yes	Yes
Combined heat and mass transfer in building envelopes				
Thermal comfort	No	Yes	Yes	Yes
Human comfort model based on activity, inside drybulb, humidity, and radiation				
Anisotropic sky model	Yes	No	No	Yes
Sky radiance depends on sun position for better calculation of diffuse solar on tilted surfaces				
Advanced fenestration calculations	Yes	No	No	Yes
Controllable window blinds				
Electrochromic glazing				
WINDOW 5 calculations				
More than 200 window types — conventional, reflective, low-E, gas-fill, electrochromic	Yes	No	No	Yes
Layer-by-layer input for custom glazing	No	No	No	Yes
Daylighting illumination and controls	Yes	No	No	Yes
Interior illuminance from windows and skylights				
Step, dimming, on/off luminaire controls				
Glare simulation and control				
Effects of dimming on heating and cooling				

## 2.2. Heat and mass balance

As noted earlier, the underlying building thermal zone calculation method in EnergyPlus is a heat balance model. The fundamental assumption of heat balance models is that air in each thermal zone can be modeled as well stirred with uniform temperature throughout. Although this does not

reflect physical reality well, the only current alternative is computational fluid dynamics (CFD) — a complex and computationally intensive simulation of fluid (in this case, air) movement. Currently, CFD is most useful in research applications. Several groups are developing models somewhere between the well-stirred model and a full CFD calculation. The modular structure of EnergyPlus allows

Table 3  
Comparison of HVAC features and capabilities

HVAC systems and equipment feature	DOE-2	BLAST	IBLAST	EnergyPlus
Fluid loops	No	No	No	Yes
Connect primary equipment and coils				
Hot water loops, chilled water and condenser loops, refrigerant loops				
Air loops	No	No	No	Yes
Connect fans, coils, mixing boxes, zones				
User-configurable HVAC systems	No	No	No	Yes
High-temperature radiant heating	No	Yes	Yes	Yes
Gas/electric heaters, wall radiators				
Low-temperature radiant heating/cooling	No	No	Yes	Yes
Heated floor/ceiling				
Cooled ceiling				
Atmospheric pollution calculation	Yes	Yes	No	Yes
CO <sub>2</sub> , SO <sub>x</sub> , NO <sub>x</sub> , CO, particulate matter and hydrocarbon production				
On-site and at power plant				
Calculate reductions in greenhouse gases				
SPARK link	No	No	No	Yes
TRNSYS link	No	No	No	Yes

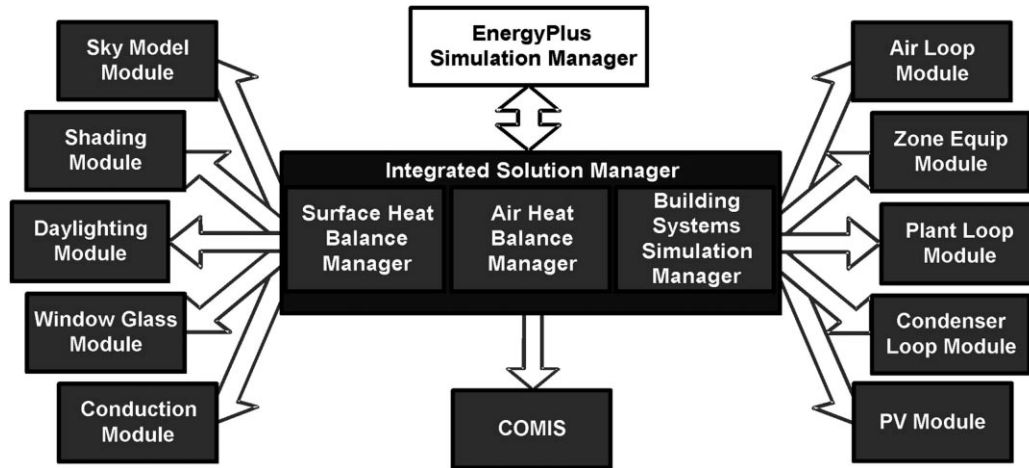


Fig. 3. Integrated simulation manager.

these new models to be included in future releases once they are available. The other major assumption in heat balance models is that room surfaces (walls, windows, ceilings, and floors) have:

- uniform surface temperatures;
- uniform long- and short-wave irradiation;
- diffuse radiating surfaces and
- one dimensional heat conduction.

Fig. 3 shows the structure of the EnergyPlus integrated solution manager that manages the surface and air heat balance modules and acts as an interface between the heat balance and the building systems simulation manager. The surface heat balance module simulates inside and outside surface heat balance; interconnections between heat balances and boundary conditions; and conduction, convection, radiation, and mass transfer (water vapor) effects. The air mass balance module deals with various mass streams, such as ventilation air, exhaust air, and infiltration. It accounts for thermal mass of zone air and evaluates direct convective heat gains. Through this module we are connecting to COMIS [8] for improved multizone airflow, infiltration, indoor contaminant, and ventilation calculations.

In addition to the basic heat and mass balance engine from IBLAST, we created three new modules based on capabilities within DOE-2: daylighting illumination [9]; WINDOW 4-based fenestration [10] and anisotropic sky. The daylighting module calculates interior daylight illuminance, glare from windows, glare control, and electric lighting controls (on/off, stepped, continuous dimming), and calculates electric lighting reduction for the heat balance module. In the future, the daylighting module will include an improved interior interreflection calculation, reflection from neighboring buildings and handling of complex fenestration systems (blinds, light shelves, roof monitors etc.). The fenestration module includes capabilities from the soon-to-be released WINDOW 5 — accurate angular dependence

of transmission and absorption for both solar and visible radiation, and temperature-dependent  $U$ -value. Users can enter a layer-by-layer window description or choose windows from the library (conventional, reflective, low-e, gas fill, electrochromic). For sun control, movable interior and exterior window shades and electrochromic glazing can be simulated. The WINDOW 5 algorithms now also account for coatings and framing elements. The sky model includes non-isotropic radiance and luminance distribution throughout the sky based on an empirical model [11,12] as a function of sun position and cloud cover. This non-uniform radiance distribution improves calculation of diffuse solar on tilted surfaces (walls and sloped roofs).

Several other modules have been reengineered for inclusion in EnergyPlus: solar shading from BLAST and conduction transfer function calculations from IBLAST. The major enhancements of the IBLAST (and EnergyPlus) heat balance engine over BLAST include mass transfer and radiant heating and cooling. The mass transfer capability within EnergyPlus allows fundamental, layer-by-layer solution for mass transfer through surfaces and a mass balance on zone air similar to the air heat balance. The radiant heating and cooling models are an expansion of the conduction transfer function and incorporate thermal comfort calculations. This provides a means for improved modeling and control capabilities for the new building systems simulation manager.

One last important feature of the EnergyPlus heat balance engine is that it is essentially identical in functionality to the Loads toolkit being developed by UI under ASHRAE Research Project 987. UI is developing both the Loads toolkit and the EnergyPlus heat and mass balance engine and is using the programming standard developed in the EnergyPlus project to produce the Loads toolkit. Both projects benefit: modularization efforts started by EnergyPlus will be useful in the Loads toolkit and new component models developed for the Loads toolkit will enhance

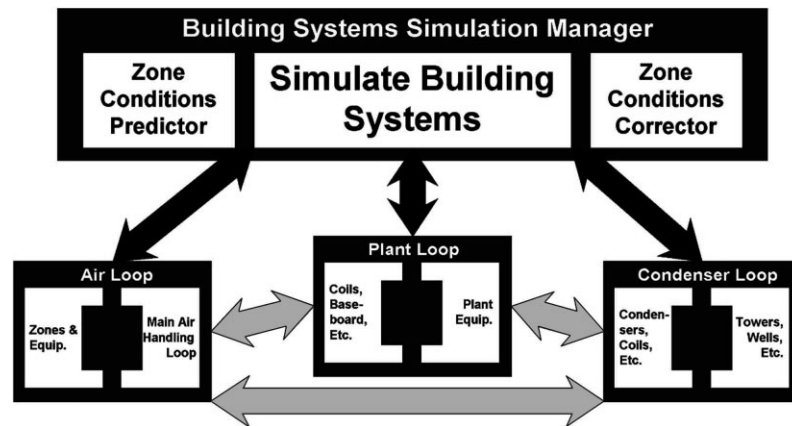


Fig. 4. Building systems simulation manager.

EnergyPlus. Strand et al. [13] provide more information on the heat and mass balance implementation in EnergyPlus.

### 2.3. Building systems simulation manager

After the heat balance manager completes simulation for a time step, it calls the building systems simulation manager, which controls the simulation of HVAC and electrical systems, equipment and components and updates the zone-air conditions (see Fig. 4). EnergyPlus does not use a sequential simulation method (first building loads, then air distribution system, and then central plant) as found in DOE-2 and BLAST since this imposes rigid boundaries on program structures and limits input flexibility. Instead, we designed the building systems simulation manager with several objectives in mind:

- fully integrated simulation of loads, systems, and plant;
- modular;
- extensible.

Integrated simulation models capacity limits more realistically and tightly couples the air and water side of the system and plant. Modularity is maintained at both the component and system level. This eases adding new components and flexibly modeling system configurations and, at the system level, equipment and systems are clearly connected to zone models in the heat balance manager. To implement these concepts, we use loops throughout the building systems simulation manager — primarily HVAC air and water loops. Loops mimic the network of pipes and ducts found in real buildings and eventually will simulate head and thermal losses that occur as fluid moves in each loop. As mentioned earlier, EnergyPlus has no hardwired ‘template’ systems. Instead, we are developing equivalent input file templates for each of the major system types in BLAST and DOE-2. These templates provide an easy starting point for users to develop inputs for system configurations that differ from ‘default’ configurations. The air loop simulates: air transport; conditioning and mixing; supply

and return fans; central heating and cooling coils; heat recovery and controls for supply air temperature and outside air economizer. The air loop connects to the zone through the zone equipment. Zone equipment includes diffusers, reheat/recool coils, supply air control (mixing dampers, fan-powered VAV box, induction unit, VAV dampers), local convection units (window air-conditioner, fan coil, water-to-air heat pump, air-to-air heat pump), higher temperature radiant/convective units (baseboard, radiators) and low-temperature radiant panels. Fig. 5 shows equipment connections to zones — note that more than one equipment type can be specified for a zone. However, users must specify equipment in the order it will be used to meet zone heating and cooling demand.

For the air loop, the solution method is iterative, not ‘single-pass’ as in DOE-2 and BLAST. In order to specify equipment connections to a loop, nodes are defined at key locations around the loop with each node assigned a unique numeric identifier as shown in Fig. 6. Node identifiers store loop state variables and set-point information for that location in the loop. We use an iterative solution technique to solve for unknown state variables along with control equation representations. These representations connect the set points at one node with the control function of a component, such as fan damper position and cooling coil water flow rate. In this schema, all the loop components are simulated first, and then the control equations are updated using explicit finite difference. This procedure continues until the simulation converges. Typical control schemes are included in the input file templates described earlier.

There are two loops for HVAC plant equipment — a primary loop (for supply equipment such as boilers, chillers, thermal storage, and heat pumps) and a secondary loop (for heat rejection equipment such as cooling towers and condensers). Fig. 7 presents a schematic view of equipment connections on the primary plant loop. Equipment is specified by type (gas-fired boiler, open drive centrifugal chiller) and its operating characteristics. In the first release of EnergyPlus, we are supporting performance-based

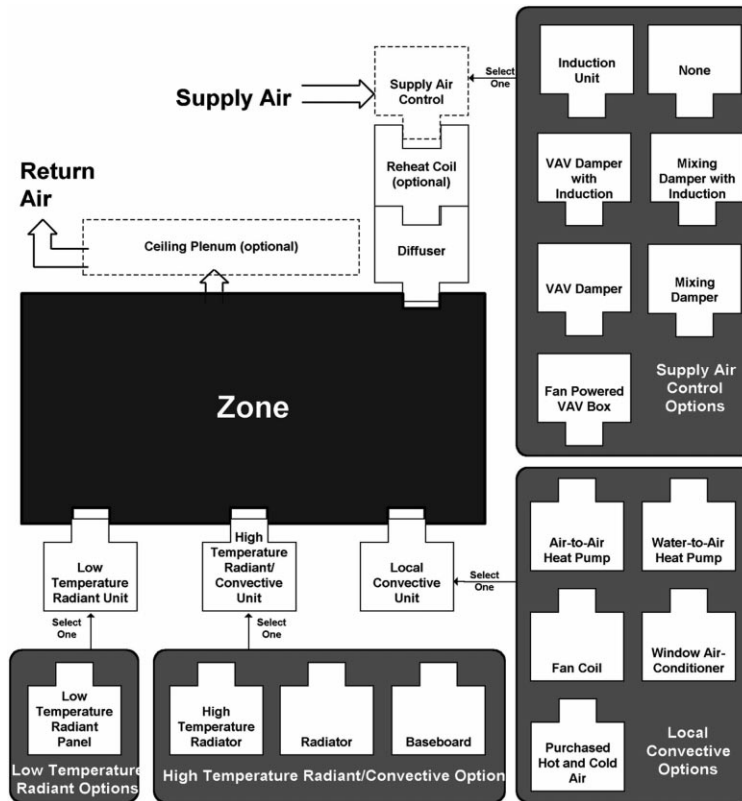


Fig. 5. Zone equipment module.

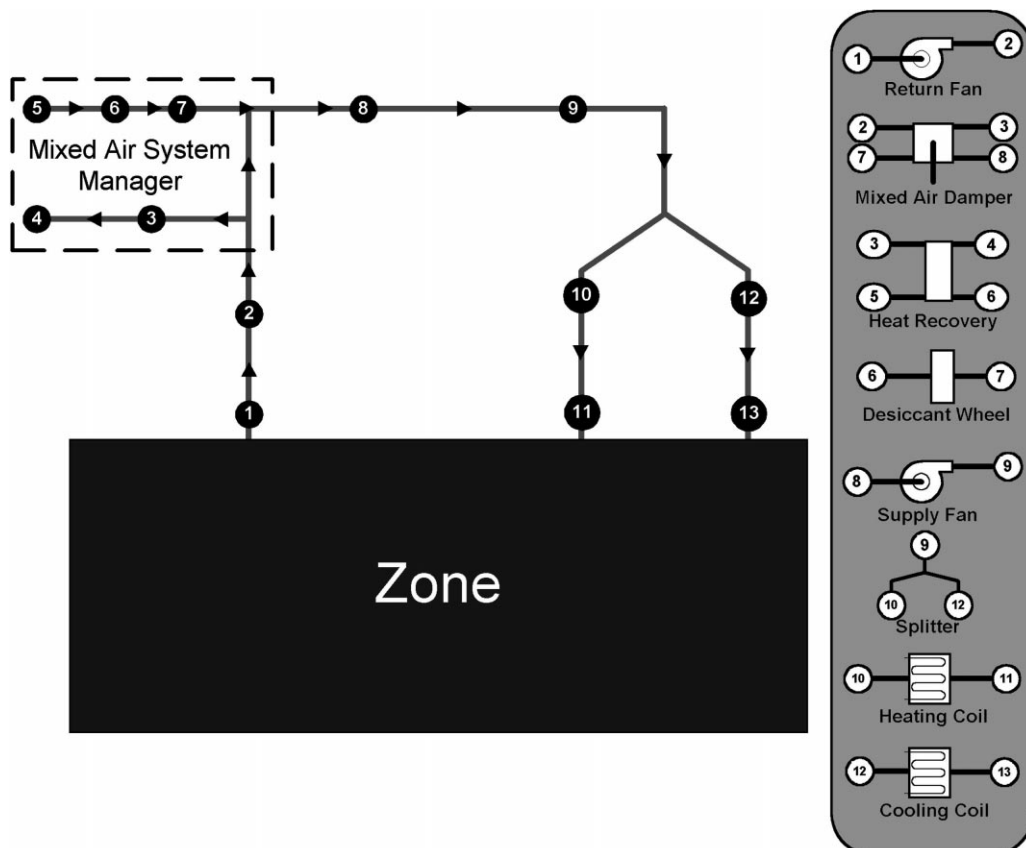


Fig. 6. Simple air loop node diagram.



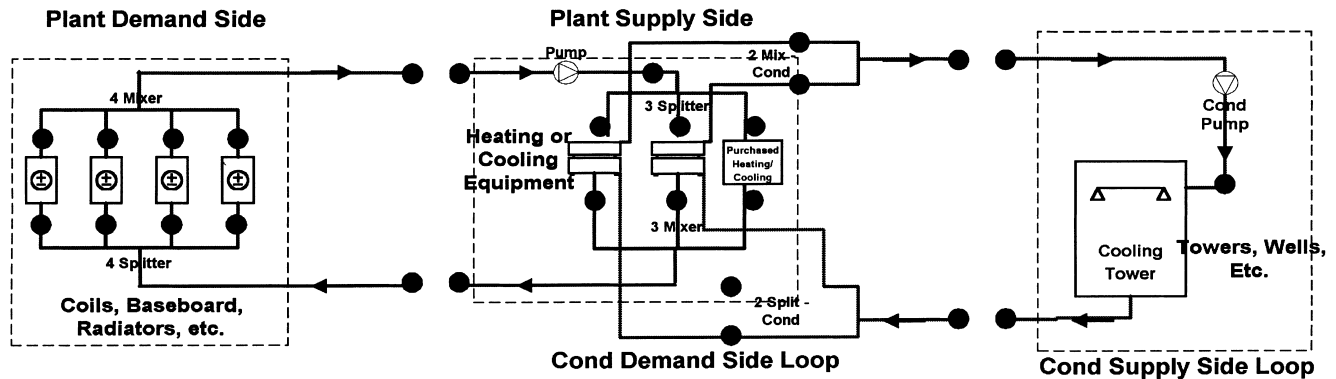


Fig. 7. Example plant loop.

equipment models (such as in BLAST and DOE-2). But because of the modular code, it will be easy for developers to add other types of models. As in the air loop, the primary and secondary plant loops use explicit nodes to connect equipment to each loop. Connections between the air loop and zone equipment and the primary and secondary loops are made through the node data structure and must be explicitly defined in the input file. Fisher et al. [14] provide additional information about the modular, loop-based approach to building systems simulation in EnergyPlus. A similar loop approach is proposed for a new electrical loop for simulating electrical systems — supply (utility, photovoltaic modules, and fuel cells), demand (plug loads, lighting, and other electrical loads), and measurement (meters).

In the longer term, EnergyPlus users will have more systems and equipment options through a link to SPARK [6], an equation-based simulation tool. SPARK is a better solver for complex iterative problems. SPARK already has a library of HVAC components based on the ASHRAE primary and secondary toolkits [15,16]. EnergyPlus will continue to have system types (in input file templates), but developers and advanced users will be able to easily build complex new HVAC models with SPARK. A similar link is under development to the TRNSYS simulation model [7], which will provide users with system/plant, photovoltaic, and solar thermal simulation models.

### 3. Input, output, and weather data

Both DOE-2 and BLAST have highly-structured, but user readable input file definitions that have evolved over many years. Instead of user readability, we designed the EnergyPlus input files for easy maintenance and expansion. We chose to keep the input file simple in order to accept simulation input data from other sources such as CADD systems, programs that also do other functions, and preprocessors similar to those written for BLAST and DOE-2. An EnergyPlus input file, while readable, is cryptic and definitely not user-friendly — it is not intended to be the main interface for typical end-users. We expect most users

will use EnergyPlus through an interface from a third-party developer. To make it easy for current DOE-2 and BLAST users to move to EnergyPlus, the team has written utilities that convert BLAST and DOE-2 input to the new EnergyPlus input structure.

Depending on how quickly the International Alliance for Interoperability (IAI) progresses in defining their building information exchange, a common object-oriented data store such as the IAI's proposed Industry Foundation Classes [17] may eventually become the main interface to the program.

EnergyPlus uses a free format input file that contains a complete object-based description of the building and its systems. The basic syntax is:

object, data, data, data, . . . , data;

'Object' is a pre-defined word denoting a building component, such as SURFACE, MATERIAL, LIGHTING, SYSTEM, HEATING COIL, and BOILER. This word is followed by a list of data values and terminates with a semicolon. These data describe performance characteristics and intended use for that object in the simulation. Unlike BLAST and DOE-2, the input file must explicitly provide all information — there are very few default assumptions. Users may include comments throughout their input data file. A comparison of input file syntax for BLAST, DOE-2, and EnergyPlus is shown in Table 4.

The input file syntax is not hardwired within EnergyPlus; instead EnergyPlus reads an input data dictionary at runtime to determine the syntax of the input data file. The general syntax of the input data dictionary is:

Object, A1 [an alpha], N1 [a number], . . . ;

For example, for the EnergyPlus Location command, the data dictionary line is:

Location, A1 [Location Name], N1 [Latitude], N2 [Longitude], N3 [Time Zone], N4 [Elevation];

This tells the input processor that, for the Location command, to expect one text field (A1) with the location name, and four numeric inputs (N1, N2, N3, and N4)-latitude, longitude, time zone and elevation, respectively. Words in

Table 4  
Comparison of BLAST, DOE-2 (BDL) and EnergyPlus input

	BLAST	BDL (DOE-2)	EnergyPlus
Location (simple input) Description	Location defined in library. Library includes name, latitude, longitude, elevation, and time zone LOCATION = Name;	Location information defined by input, defaults to information on weather file BUILDING-LOCATION, LATITUDE = W, LONGITUDE = X, ALTITUDE = Y, TIME-ZONE = Z..	Location information defined by input Location, Name, Latitude, Longitude, Elevation, Time-Zone;
Input syntax			
Example input	LOCATION = CHICAGO;	BUILDING-LOCATION, LATITUDE = 41.98, LONGITUDE = 87.90, ALTITUDE = 673, TIME-ZONE = 6..	LOCATION, Chicago Illinois USA, 41.98, -87.90, -6, 205;
Material (more complex) Description	Material defined in library. Library includes material name, conductivity, density, specific heat, resistance, roughness, and moisture properties	Material from library or defined in input, includes thickness, conductivity, density, specific heat, or resistance. Thickness restated during layer input (optional) Material = A, Thickness = W, Conductivity = X, Density = Y, Specific Heat = Z..	All material information defined by input
Input syntax	TEMPORARY MATERIAL: Usname = (L = usn1, K = usn2, CP = usn3, D = usn4, ABS = usn5, TABS = usn6, R = usn7, TRANS = usn8, IR = usn9, FILMTRANS = usn10, REF=usn11, SC = usn12, roughness, asg); END; Brick = (L = 0.3333, K = 5.6, CP = 0.19, D = 120, ROUGH);		Material, Name, Thickness, Conductivity, Density, Specific Heat, Roughness;
Example input		BRICK = MATERIAL, THICKNESS = 0.3333, CONDUCTIVITY = 5.6, DENSITY = 120, SPECIFIC-HEAT = 0.19..	MATERIAL, Brick, 0.1016, 0.721, 1922, 837, Medium;

brackets [ ] describe the variable and its units (meters, l/s, etc.).

During a simulation, EnergyPlus saves results for each time step in an output data structure and these results can be reported at each time step or aggregated to longer time intervals. This structure and subsequent output uses a similar philosophy to the input — simple text files with a syntax of object, time stamp, data, data, data, . . . , data;. The output data is simple yet contains much of the simulation results so that users and interface developers can easily access specific results without modifying the calculation engine.

Several report types are available — standard output (user specified variables at specified time intervals), one time output (such as input echo, input verification and interim calculations), and visual surface output (including DXF outputs). Because the data structure is simple and comma-separated, output post-processors can easily read the data and create more elaborate reports. One drawback of our simple file format is that the output files can become very large.

The other major data input is weather. Rather than a binary file created by a separate weather processor, again we use a simple text-based format, similar to the input data and output data files. The weather data format includes basic location information in the first eight lines: location (name, state/province/region, country), data source, latitude, longitude, time zone, elevation, peak heating and cooling design conditions, holidays, daylight savings period, typical and extreme periods, two lines for comments, and period covered by the data. The data are also comma-separated and contain much of the same data in the TMY2 weather data set [18]. EnergyPlus does not require a full year or 8760 (or 8784) h in its weather files. In fact, EnergyPlus allows and reads subsets of years and even sub-hourly (5 min, 15 min) data — the weather format includes a ‘minutes’ field. EnergyPlus comes with a utility that reads standard weather service file types such as TD1440 and DATSAV2 and newer ‘typical year’ weather files such as TMY2, IWECC, and WYEC2. More information on the weather data format is contained in [19].

In summary, all the data files associated with EnergyPlus — input, output, and weather — have simple self-contained formats but they can become quite large. The data files can be easily read and interpreted by other programs — spreadsheets, databases, or custom interface programs. By working with third party interface developers early, we will keep these files simple and easy to use by other programs that building designers use.

#### 4. Adding a new module

One of the main goals for EnergyPlus is to make it easy for developers to add new features and modules. The process is conceptually simple. Reference the following generic steps:

- Find (or develop your own) model: define the model parameters, equations, specialized coefficients and needed data.
- Modify for loop structure: determine the “plug-in” point, where the module will be called within EnergyPlus.
- Write input specs: determine the inputs that will be necessary to “execute” the model. Envision how this will be put into the EnergyPlus Input Data Dictionary/Input Data File structure.
- Write new module(s); add call from higher-lever manager: using the EnergyPlus Programming Standard, and other developer documents, write the code — possibly breaking simulation tasks into modules.
- Write GetInput: create the actual syntax in the Input Data Dictionary and produce the proper “GetInput” routine in the new module to obtain the data.
- Finish the simulation routine.
- Debug simulation routine; resolve snags associated with interactions with other loops.
- Implement report variables; use to recheck results, repair model.

#### 5. Testing

A critical part of EnergyPlus development has been continuous testing using several simultaneous paths. The primary emphasis to date has been on comparative and analytical testing. Comparative testing of basic loads algorithms has been completed using the ASHRAE Standard 140P [20] series of tests, which consist of a basic shoebox with windows and shading for both low mass and high mass construction. Fig. 8 shows sample EnergyPlus results compared against reference data for a number of other simulation programs provided with draft Standard 140P. Analytical tests in progress include the BEPAC conduction tests [21], which provide analytical solutions for conduction through a range of thicknesses for a variety of materials from aluminum to concrete. The conduction is driven by step, sine and ramp exterior temperature functions. Comparative version testing of EnergyPlus against itself has been extremely useful in detecting and resolving problems introduced during development. The testing to date demonstrates that EnergyPlus provides results in good agreement with other simulation programs for simple cases. Testing will continue throughout the development of EnergyPlus and the results and methodologies will be made available to users as part of the EnergyPlus documentation.

#### 6. Release 1.0 and beyond

The initial working version of EnergyPlus, or alpha version, was completed in December 1998 for internal testing by the team. In May 1999, a first internal version for testing became available to outside developers. In December 1999, the first beta test version of EnergyPlus

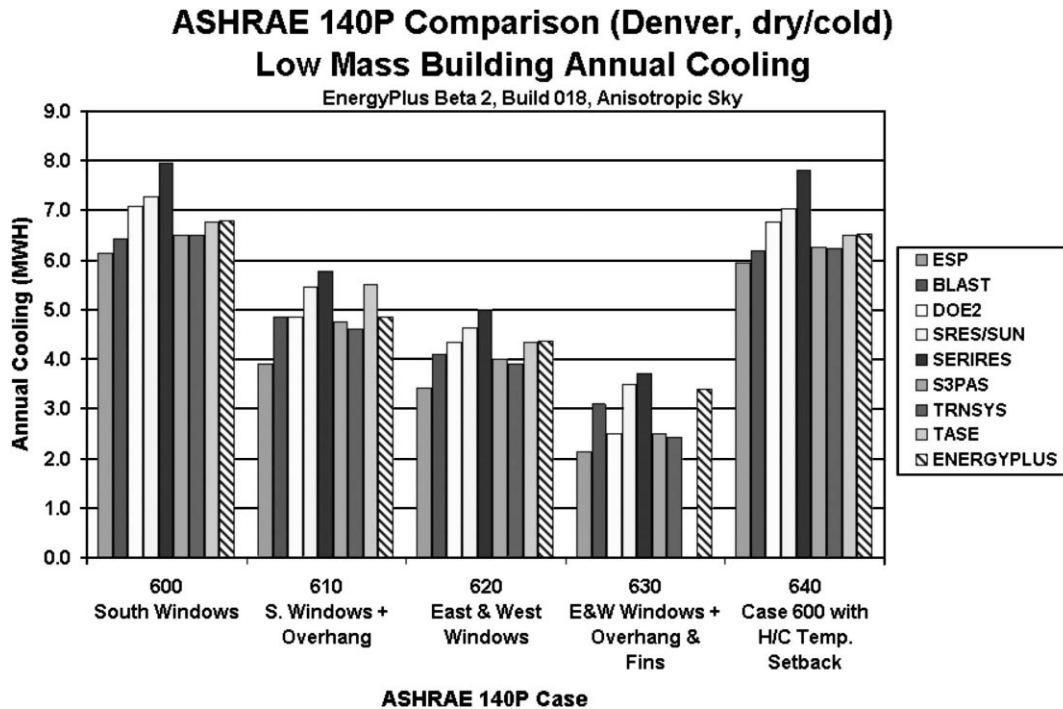


Fig. 8. Sample EnergyPlus comparative testing results.

was released for general testing by potential users. Depending on progress in testing and debugging through a series of five beta versions, we plan to release version 1.0 of EnergyPlus in early 2001.

In late 2000, we will begin planning for the second release of EnergyPlus based on new features suggested by users, developers, and the team. Working with a coordinating group of users and developers, we will then select the features and capabilities for that release. We plan to release updates to EnergyPlus on an 18-month release cycle. Some new features already under development are a connection to the COMIS airflow program, improved ground heat transfer, electrical system simulation, and solar thermal and photovoltaic modules. The link to COMIS will allow better calculation of infiltration, natural ventilation, multizone airflow, and air pollutant transport. The ground heat transfer model will either be a 2- or 3-dimensional heat transfer calculation for various foundation configurations.

## 7. Summary

EnergyPlus is a new building energy simulation program that builds on the strengths of BLAST and DOE-2. It is being written in Fortran 90 with structured, modular code that is easy to maintain, update, and extend. Benefits of EnergyPlus include:

For simulation program users:

- limits inherent in BLAST and DOE-2 (such as number of zones, schedules, or systems) are eliminated by the new

structures in EnergyPlus — now limited only by a user's computer resources rather than hardwired in code;

- EnergyPlus source code is open for inspection — and understandable;
- developers around the world will be able to develop new modules — algorithmic or interfaces;
- new module development can keep pace with new building technologies, maximizing public impact of latest buildings research.

For simulation developers:

- standardized structure significantly decreases the learning curve for developers;
- new, structured, modular code is easier to understand and work with;
- modular structure allows developers to work in parallel on new modules.

General benefits include:

- simulation capabilities include integrated simulation, combined heat and mass transfer balance, multizone airflow, HVAC loops (flexible system and plant simulation), links to SPARK and TRNSYS system/plant simulation, and algorithms from the new ASHRAE loads toolkit;
- input, output, and simulation capabilities are much more flexible;
- EnergyPlus will be released quicker than adding similar capabilities to either DOE-2 or BLAST, but offers similar benefits.

Although the two workshops sponsored by DOE and DOD [5] pointed up the critical need for good user interfaces in the

success of any simulation tool, the EnergyPlus team is focusing first on developing the heart of a new simulation tool — the calculation engine. We consciously incorporated the priorities of the workshop participants in our development effort (many can be seen in Fig. 2). The EnergyPlus team that is working with third party interface developers to ensure user-friendly interfaces and other new modules are ready when the program is released.

EnergyPlus not only combines the best features of the BLAST and DOE-2 programs, but also represents a significant step forward in terms of computational techniques and program structures. Connectivity and extensibility are overriding objectives in the design and development process. This will ensure broad participation in program enhancement and facilitate third party interface and module development. EnergyPlus beta testing began in early 1999. The first release is scheduled for early 2001. Up to date information on EnergyPlus, is provided on the EnergyPlus web site (see web resources below).

## 8. Web resources

Information on EnergyPlus, including schedule, availability of beta releases, documentation, licensing, programming standards, and other documentation: [http://www.eren.doe.gov/buildings/energy\\_tools/energyplus.htm](http://www.eren.doe.gov/buildings/energy_tools/energyplus.htm).

Workshops on next generation energy simulation tools: [http://www.eren.doe.gov/buildings/energy\\_tools/workshops.htm](http://www.eren.doe.gov/buildings/energy_tools/workshops.htm).

Web-based directory of more than 200 building-related software tools from around the world: [http://www.eren.doe.gov/buildings/tools\\_directory/](http://www.eren.doe.gov/buildings/tools_directory/).

## References

- [1] Building Systems Laboratory, BLAST 3.0 Users Manual, Department of Mechanical and Industrial Engineering, Building Systems Laboratory, University of Illinois, Urbana-Champaign, IL, 1999.
- [2] F.C. Winkelmann, B.E. Birdsall, W.F. Buhl, K.L. Ellington, A.E. Erdem, J.J. Hirsch, S. Gates, DOE-2 Supplement, Version 2.1E, LBL-34947, Lawrence Berkeley National Laboratory, National Technical Information Service, Virginia, Springfield, November 1993.
- [3] R.D. Taylor, C.O. Pedersen, L.K. Lawrie, Simultaneous simulation of buildings and mechanical systems in heat balance based energy analysis programs, in: Proceedings of the 3rd International Conference on System Simulation in Buildings, Liege, Belgium, 3–5 December, 1990.
- [4] R.D. Taylor, C.O. Pedersen, D.E. Fisher, R.J. Liesen, L.K. Lawrie, Impact of simultaneous simulation of building and mechanical systems in heat balance based energy analysis programs on system response and control, in: Proceedings of Building Simulation'91, IBPSA, Sophia Antipolis, France, August 1991.
- [5] D.B. Crawley, L.K. Lawrie, F.C. Winkelmann, W.F. Buhl, A.E. Erdem, C.O. Pedersen, R.J. Liesen, D.E. Fisher, The next-generation in building energy simulation — a glimpse of the future, in: Proceedings of Building Simulation'97, IBPSA, Vol. II, Prague, Czech Republic, September 1997, pp. 395–402.
- [6] W.F. Buhl, A.E. Erdem, F.C. Winkelmann, E.F. Sowell, Recent improvements in SPARK: strong-component decomposition, multi-valued objects and graphical editors, in: Proceedings of the Building Simulation'93, IBPSA, Adelaide, South Australia, Australia, August 1993, pp. 283–289.
- [7] Solar Energy Laboratory, TRNSYS Version 15 User Manual and Documentation, Solar Energy Laboratory, Mechanical Engineering Department, University of Wisconsin, Madison, WI, February 2000.
- [8] Y.J. Huang, F.C. Winkelmann, W.F. Buhl, C.O. Pedersen, D.E. Fisher, R.J. Liesen, R. Taylor, R.K. Strand, D.B. Crawley, L.K. Lawrie, Linking the COMIS multi-zone airflow model with the EnergyPlus building energy simulation program, in: Proceedings of Building Simulation'99, IBPSA, Vol. II, Kyoto, Japan, September 1999, pp. 1065–1070.
- [9] F.C. Winkelmann, S.E. Selkowitz, Daylighting simulation in the DOE-2 building energy analysis program, *Energy and Buildings* 8 (1985) 271–286.
- [10] D.K. Arasteh, E.U. Finlayson, C. Huizenga, Window 4.1: Program Description, Report no. LBL-35298, Lawrence Berkeley Laboratory, Berkeley, CA, 1994.
- [11] R.R. Perez, R. Seals, J. Michalsky, R. Stewart, Modeling daylight availability and irradiance components from direct and global irradiance, *Solar Energy* 44 (1990) 271–289.
- [12] R.R. Perez, P. Ineichen, E. Maxwell, F. Seals, A. Zelenda, Dynamic models for hourly global-to-direct irradiance conversion, in: Proceedings of the Biennial Congress of the International Solar Energy Society, 1991 Solar World Congress, Vol. 1, Part II, 1991, pp. 951–956.
- [13] R. Strand, F. Winkelmann, F. Buhl, J. Huang, R. Liesen, C. Pedersen, D. Fisher, R. Taylor, D. Crawley, L. Lawrie, Enhancing and extending the capabilities of the building heat balance simulation technique for use in EnergyPlus, in: Proceedings of Building Simulation'99, IBPSA, Vol. II, Kyoto, Japan, September 1999, pp. 653–660.
- [14] D.E. Fisher, R.D. Taylor, F. Buhl, R.J. Liesen, R.K. Strand, A modular, loop-based approach to HVAC energy simulation and its implementation in EnergyPlus, in: Proceedings of Building Simulation'99, IBPSA, Kyoto, Japan, September 1999.
- [15] M.J. Brandemuehl, G. Shauna, A. Inger, HVAC2 Toolkit: Algorithms and Subroutines for Secondary HVAC System Energy Calculations, American Society of Heating, Refrigerating and Air-conditioning Engineers, Atlanta, GA, 1993.
- [16] J. LeBrun, J.-P. Bourdouxhe, M. Grodent, HVAC1 Toolkit: A Toolkit for Primary HVAC System Energy Calculation, American Society of Heating, Refrigerating, and Air-conditioning Engineers, Atlanta, GA, 1999.
- [17] V. Bazjanac, D. Crawley, Industry foundation classes and interoperable commercial software in support of design of energy-efficient buildings, in: Proceedings of the Building Simulation'99, IBPSA, Vol. II, Kyoto, Japan, September 1999, pp. 661–668.
- [18] National Renewable Energy Laboratory (NREL), User's Manual for TMY2s (Typical Meteorological Years), NREL/SP-463-7668, and TMY2s, Typical Meteorological Years Derived from the 1961–1990 National Solar Radiation Data Base, CD-ROM, NREL, Golden, CO, June 1995.
- [19] D.B. Crawley, J.W. Hand, L.K. Lawrie, Improving the weather information available to simulation programs, in: Proceedings of Building Simulation'99, IBPSA, Vol. II, Kyoto, Japan, September 1999, pp. 529–536.
- [20] ASHRAE, BSR/ASHRAE Standard 140P, Standard Method of Test of the Evaluation of Building Energy Analysis Computer Programs, first public review draft, ASHRAE, Atlanta, GA, April 2000.
- [21] B.H. Bland, Conduction tests for the validation of dynamic thermal models of buildings, BEPAC Technical Note 93/1, BEPAC, August 1993.